



VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS INSTITUTAS
KOMPIUTERINIO IR DUOMENŲ MODELIAVIMO KATEDRA

Bakalauro baigiamasis darbas

Navigacija patalpose naudojant Bluetooth stotelių ir išmaniojo įrenginio jutiklių duomenis

Atliko:

Domantas Alenskas

parašas

Vadovas:

jaun. asist. Tomas Raila

Vilnius
2020

Turinys

Santrauka	4
Summary	5
Ivyadas	6
1. Bluetooth stotelės	7
1.1. Bluetooth Low Energy	7
1.2. RSSI indikatorius	7
1.3. Bluetooth low energy stotelės	8
1.4. Navigacijos patalpose metodai	8
1.4.1. Diapozono pagrindu pagrįstas metodas	8
1.4.2. Piršto antspaudo metodas	8
2. Išmaniojo įrenginio duomenys	9
2.1. Nuokrypio išlyginimas	9
2.2. Sukimosi vektoriaus jutiklis	9
2.3. Magnetinio lauko jutiklis ir akselerometras	10
2.4. Posūkio nustatymas	10
3. Kompiuterio mokymas	11
3.1. Kompiuterio mokymo modelis	11
3.2. Kompiuterio mokymo metodai	11
3.2.1. Prižiūrimas mokymas	11
3.2.2. Neprižiūrimas mokymas	11
3.2.3. Skatinamasis mokymas	12
3.3. Duomenų dalinimas	12
3.4. Duomenų normalizacija	12
3.5. Klasifikacija	13
3.5.1. Naiviojo Bajeso klasifikatorius	13
3.5.2. Atraminių vektorių klasifikatorius	14
3.6. Regresija	14
3.6.1. K-artimiausio kaimyno regresija	15
3.6.2. Ridge regresija	15
4. Tyrimas	16
4.1. Duomenys ir jų rinkimas	16
4.1.1. BLE stotelių duomenys	16
4.1.2. Kompaso duomenys	17
4.2. Kompiuterio mokymo eiga ir rezultatai	18
4.2.1. Kompaso duomenų įtaka	19
4.2.2. Duomenų normalizavimo įtaka	20
4.2.3. Regresijos tikslumo rodiklių analizė	21
4.3. Tyrimo rezultatai	22
Išvados ir rekomendacijos	23

Ateities tyrimų planas	24
Literatūros šaltiniai	25

Santrauka

Augant patalpų dydžiui ir sudėtingumui išauga patogaus būdo naviguoti tokiose patalpose poreikis. Navigacija patalpose gali būti įgyvendinta naudojant Bluetooth stoteles ir išmanaus įrenginio jutiklius. Šiame darbe yra analizuojamas minėtas navigacijos būdas patalpose naudojant sudaromą pirštų antspaudų žemėlapių pagal turimus ir renkamus duomenis. Pagrindinis darbo tikslas yra išsiaiškinti išmanaus įrenginio jutiklių įtaką navigacijai patalpose. Tam yra naudojama Python programavimo kalba ir kompiuterio mokymui skirta biblioteka "scikit-learn". Darbe analizuojami ir lyginami klasifikacijos ir regresijos modelių koordinačių spėjimų tikslumas tiek su kompasu duomenimis, tiek be jų. Tai pat yra tiriama duomenų normalizavimo įtaka. Atlikus tyrimą, buvo padarytos išvados, jog koordinatėms nustatyti geriausia yra naudoti k-artimiausio kaimyno regresijos algoritmą. Spėjant ilgumą, kompasu duomenys naudos nedavė, tačiau spėjant plotumą, nauda buvo reikšminga. Regresijos algoritmams duomenų normalizavimas spėjimų tikslumui naudos nesuteikė.

Summary

Indoor Navigation Using Bluetooth Beacons and Smart Device's sensors data

As the size and complexity of the premises increases, the need for a convenient way to navigate in such premises increases. Indoor navigation can be accomplished using Bluetooth stations and smart device sensors. This work analyzes the aforementioned indoors navigation method using a fingerprint map based on available and collected data. The main purpose of this work is to find out the influence of smart device sensors for indoor navigation. Python programming language and the "scikit-learn" machine learning library were used to achieve that. The work analyzes and compares the accuracy of the classification and regression model coordinate predictions with and without compass data. The influence of data normalization is also investigated. The study concluded that it is best to use the k-nearest neighbor regression algorithm to determine the coordinates. There was no benefit in using compass data to predict longitude, but when predicting latitude, the benefits were significant. Regression algorithms did not benefit from data normalization for prediction accuracy.

Ivadas

Šiomis dienomis, navigacija išmaniuosiuose įrenginiuose yra neatsiejama kiekvienos dienos dalis. Globali padėties nustatymo sistema padeda nustatyti apytiksliai įrenginio vietą žemėlapyje, nesvarbu, ar tai būtų pastatas, kelias ar plynas laukas. Deja, tačiau ši sistema negali nurodyti tikslios vietos patalpoje, kadangi ji nėra supažindinti nei su patalpos išplanavimu, nei su esančiu aukštų skaičiumi. Navigacijos patalpose nauda yra pakankamai akivaizdi: patogesnis apsipirkimas dideliuose prekybos centruose, greitesnis reikiamo kabineto suradimas ligoninėse, muziejaus eksponatų išdėstymo žinojimas. Šiai problemai yra pasiūlyti keli sprendimo būdai. Juos sudaro navigacija patalpose naudojant NFC žymeklius [7], Wi-Fi technologiją [22], magnetinius žemėlapius [6].

Vienas iš šios problemos sprendimų yra navigacija patalpose naudojant Bluetooth švyturėlius. Iš švyturėlių yra siunčiami signalai, kurių stiprumas gali būti užfiksuotas RSSI indikatoriaus reikšmėmis. Pagal minėtas reikšmes yra sudaromas pirštų antspaudų žemėlapis, pagal jį yra nustatoma nežinoma pozicija. Šiuo atveju tam yra naudojami prižiūrimo kompiuterio mokymo klasifikacijos ir regresijos algoritmai naudojant Python programavimo kalbą ir "scikit-learn" biblioteką.

Tyrimo tikslas yra išsiaiškinti, ar išmanaus įrenginio poziciniai ir inerciniai jutikliai gali patikslinti įrenginio vietą patalpoje. Iš išmanaus įrenginio jutiklių yra gaunamos reikšmės, iš kurių yra nustatoma kompasos reikšmė. Ši kompasos reikšmė prijungiama prie pirštų antspaudų žemėlapiu su prielaida, jog per tašką žemėlapyje yra einama pirmyn arba atgal su 10° paklaida. Darbo uždavinius sudaro egzistuojančios duomenų rinkimo programėlės pakeitimai, jog ši galėtų fiksuoti kompasos reikšmes, kompiuterio mokymo metodų įgyvendinimas ir panaudojimas bei kompasos duomenų naudos tyrimas.

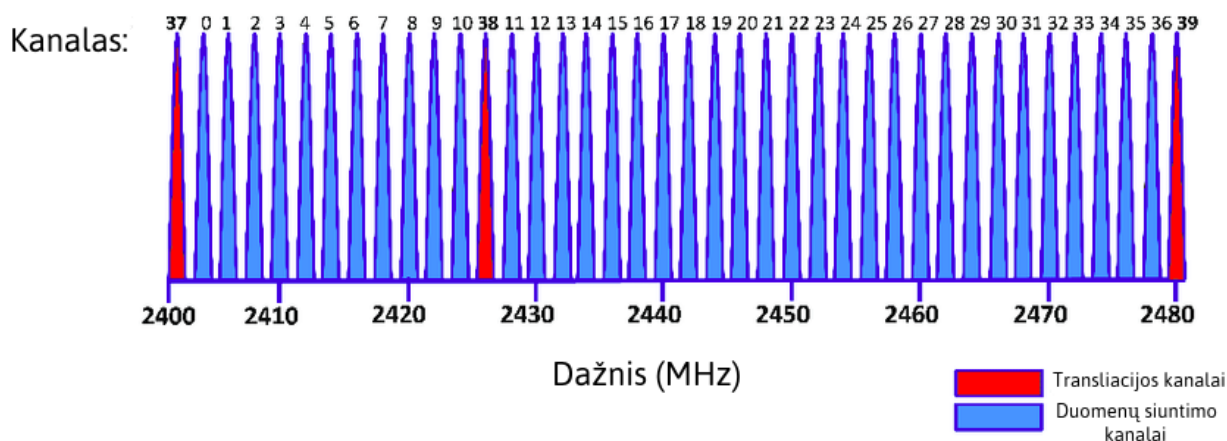
Šį darbą sudaro keturios pagrindinės dalys. Pirmoje dalyje yra aprašomos Bluetooth stotelės ir navigacijos patalpose metodai. Antroje dalyje rašoma apie išmaniojo įrenginio jutiklius ir kaip iš jų yra išgaunamos kompasos reikšmės. Trečioje dalyje aprašomi kompiuterio mokymas ir naudojami metodai. Ketvirtoje dalyje yra aprašytas atliktas tyrimas ir lyginami gauti rezultatai. Padėka Doc., Dr. Valdui Rapševičiui už šioje dalyje naudojamus jo surinktus duomenis bei jo sukurtą Android aplikaciją, kurie stipriai prisidėjo prie tyrimo eigos.

Tyrimo metu buvo išsiaiškinta, jog navigacijos patalpose problemai spręsti kartu su išmanaus įrenginio jutiklių duomenimis labiau tinkami yra regresijos tipo metodai. Tai pat buvo išsiaiškinta, jog esant per mažam Bluetooth švyturėlių kiekiui arba prastam jų išdėstymui kompasos duomenys pagerina navigaciją patalpose.

1. Bluetooth stotelės

1.1. Bluetooth Low Energy

Bluetooth Low Energy (toliau BLE) yra belaidžio asmeninio tinklo technologija. Lyginant su klasikine Bluetooth technologija, BLE yra skirtas gerokai sumažinti energijos sunaudojimą, išlaikant panašų bendravimo diapozoną. BLE duomenų perdavimas gali būti naudojamas kaip vienpusė komunikacija. Pastarieji siunčia duomenis nustatytu stiprumu ir intervalu keliais skirtingais kanalais. Signalų stiprumas gali siekti net iki 30 metrų, tačiau įprastinis veikimo diapozonas yra nuo 2 iki 5 metrų. Perdavimo jėga yra matuojama dBm. Siunčiamose žinutėse kartu yra siunčiamas unikalus siuntėjo identifikavimo raktas, kuris padeda atpažinti siuntėjo tapatybę [10]. BLE operuoja 2,4 gigahercų juostoje, taip pat kaip ir 2,4 gigahercų Wi-Fi. BLE turi 40 skirtingų kanalų, kiekvienas yra 2 megahercų pločio. Tik 3 iš šių kanalų yra pritaikyti transliacijai, kiti likę 37 yra pritaikyti siųsti duomenis ir palaikyti dvipusę komunikaciją tarp įrenginių. Kanalų schema yra pavaizduota 1 pav. Navigacijos atveju, reikalingi tik 3 transliavimo kanalai, kadangi reikalingi tik 2 dalykai - identifikavimo raktas bei signalo stiprumas. Turint juos, galima identifikuoti signalo siuntėją ir kartu žinoti, kaip toli yra siuntėjas.



1 pav. BLE kanalų schema.

1.2. RSSI indikatorius

RSSI (angl. Receiver signal strength indication) indikatorius nurodo gauto signalo stiprumą. Naudojantis šio indikatoriaus reikšme galima nustatyti, kaip toli yra siuntėjas. RSSI indikatorius gali stipriai pasikeisti net nuo mažiausio erdvinio pasikeitimo. Kiekvienas transliavimo kanalas yra siauro pločio, todėl signalas yra linkęs greit išblukti. Signalui įtaką daro ir patalpos, kurios būna itin sudėtingos (stalai, kėdės, kiti įvairūs daiktai) bei turinčios daug paviršių (sienos, keli aukštai) [30]. Vieno skenavimo metu gali būti užfiksuojami keli RSSI indikatoriai arba neužfiksuojamas nei vienas. Kadangi transliacija yra atliekama vienu metu per tris kanalus ir jeigu skenavimo intervalas yra didesnis negu transliavimo intervalas, gali būti užfiksuojami keli RSSI indikatoriai iš to paties siuntėjo. Skenavimo metu dėl signalo silpnumo ar aplinkos triukšmo indikatorius gali būti neužfiksuotas.

1.3. Bluetooth low energy stotelės

BLE stotelė yra siųstuvas, kuris siunčia savo unikalų identifikavimo raktą į aplinkinius Bluetooth priimančius įrenginius. Šios stotelės gali būti naudojamos įvairių reklaminių pranešimų ar informacijai apie netoli esančią lankytiną vietą siuntimui į išmaniuosius telefonus, navigacijai patalpose. Lyginant su panašiomis technologijomis, BLE stotelė yra pranaši tuo, jog jai nėra reikalinga speciali infrastruktūra, vidinės baterijos išlaiko įrenginio veikimą iki pusės metų, įrengimo kaina yra maža [14]. Šiuo metu rinkoje yra populiariausi du stotelių protokolai: Eddystone ir iBeacon. Eddystone yra sukurtas Google, o iBeacon - Apple, tačiau abu protokolai yra pritaikyti tiek Android, tiek iOS išmaniųjų įrenginių operacinėms sistemoms. Įprastai, navigacijai patalpose naudojamos stotelės operuoja reklamos režimu, siunčia trumpas žinutes pritaikančiu intervalu.

1.4. Navigacijos patalpose metodai

1.4.1. Diapozono pagrindu pagrįstas metodas

Diapozono pagrindu pagrįstas metodas (angl. Range-based method) naudoja RSSI indikatoriaus reikšmės atstumui nuo stotelių aptikti. Darant prielaidą, jog buvo gauti bent 3 RSSI indikatoriai, naudotojo įrenginio pozicija gali būti nustatyta naudojant trianguliaciją pagal atitinkamai apskaičiuotus atstumus. Autoriai [26] pasiūlė tris skirtingus trianguliacijos būdus.

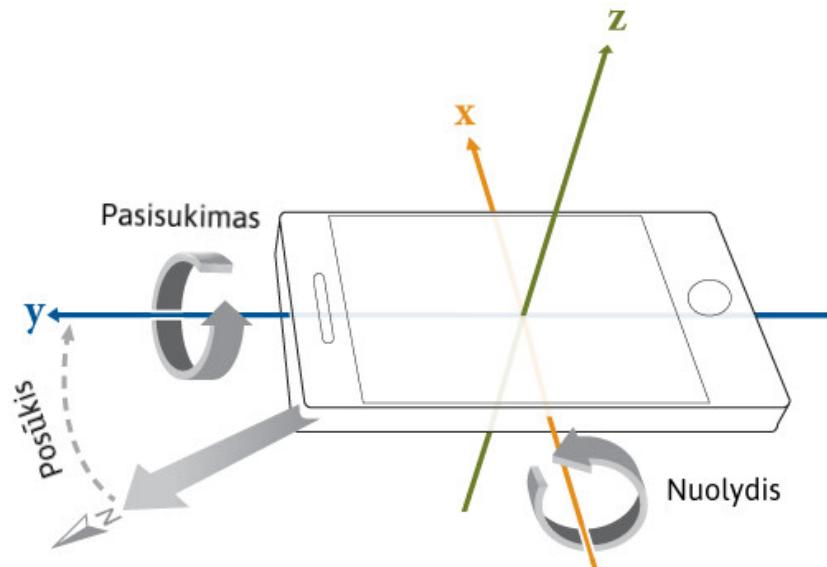
- Trijų kraštinių metodas. Nustatant lygtis, kurios vaizduoja atstumą tarp vartotojo ir siųstuvo, vartotojo padėtis gali būti surasta sprendžiant lygtis.
- Centroido metodas. Visų pirma yra apibrėžiamas poligonas pagal viršūnės, gautas iš atstumo arkų sankirtų. Tuomet, poligono centroidas yra laikomas vartotojo pozicija.
- Mažiausio kvadratinio įvertinimo (angl. Least square estimation) metodas. Sumažinus kvadratinės atstumo paklaidų sumas, galima rasti optimalią padėtį.

1.4.2. Piršto antspaudų metodas

Piršto antspaudų metodą (angl. Fingerprint-based method) sudaro dvi dalys: pasyvi ir aktyvi. Pasyviojoje dalyje, yra surenkami pirštų antspaudai. Pirštų antspaudai yra surinktos RSSI reikšmės iš BLE stotelių tam tikrame koordinatinių taške. Iš surinktų pirštų antspaudų yra sudaromas pirštų antspaudų orientacinis žemėlapis. Aktyviojoje dalyje, nežinomojo vietoje yra gaunamas piršto antspaudas, šis yra lyginamas su pastaruoju žemėlapiu siekiant išsiaiškinti nežinomą poziciją. Ši dalis yra ganėtinai paprasta, pozicijos išsiaiškinimui dažniausiai yra naudojamas k-artimiausio kaimyno metodas (angl. k-nearest neighbour). Lyginant su aktyviaja dalimi, pasyviojoje dalyje darbo krūvis yra gerokai didesnis, kadangi priimtinam tikslumui yra reikalingas didelis kiekis pirštų antspaudų. Be to, reikia tiksliai žinoti, kokiose koordinatėse buvo užfiksuotas piršto antspaudas. Pavyzdyje, pateiktame [31], antspaudai yra surenkami rankiniu būdu daugybėje fiksuotų vietų, surinkti antspaudai yra interpoliuojami pagal Gauso proceso regresijos modelį pirštų antspaudų žemėlapiui sudaryti.

2. Išmaniojo įrenginio duomenys

Vienas iš būdų navigacijai patalpose naudojant Bluetooth patikslinti yra įrenginio pozicinių ir inercinių duomenų rinkimas. Naudojant ankščiau minėta piršto antspaudo metodą tiek pasyviojoje, tiek aktyviojoje dalyse pastarieji duomenys gali pagerinti navigacijos tikslumą. Iš apdorotų duomenų galima gauti ganėtinai tikslius kompasu duomenis. Tam gauti yra reikalingas sukimosi vektoriaus jutiklis arba magnetinio lauko jutiklis kartu su akselerometru [24]. Iš jutiklio parodymų yra gaunamos trys reikšmės, nurodančios telefono padėtį erdvėje: nuolydį (x), pasisukimą (y) ir posūkį (z) (2 pav.)



2 pav. Įrenginio pozicijos ašys.

2.1. Nuokrypio išlyginimas

Tam, kad ankščiau minėtos trys reikšmės būtų tikslesnės ir stabilios, jas reikia išlyginti. Tai sumažina nuokrypį nuo prieš tai buvusios reikšmės ir taip normalizuoja duomenis [8]. Funkcijai yra reikalingi trys parametrai: prieš tai buvusi reikšmė, dabartinė reikšmė bei α reikšmė, nurodanti išlyginimo faktorių.

$$R = V_{n-1} + \alpha * (V_{n-1} - V_n), \quad 0 < \alpha < 1 \quad (2.1)$$

α turi būti tarp 0 ir 1 ir negali būti nei 0, nei 1, nes tuomet $V_{n-1} = V_n$. Toks rezultatas niekada nesikeis ir duomenys bus neteisingi.

2.2. Sukimosi vektoriaus jutiklis

Sukimo vektoriaus jutiklis (angl. Rotation vector sensor) yra virtualus jutiklis, kuris apjungia akselerometro, geometrinio lauko jutiklio bei (jeigu yra) giroskopo duomenis [3]. Visų pirma, gauti jutiklio duomenys yra išlyginami naudojant nuokrypio išlyginimo funkciją. Toliau išlyginti duomenys yra apdorojami naudojant Android operacinėje sistemoje integruotą *getRotationMatrixFromVector*, kuris priima 9 arba 16 realiųjų skaičių dydžio masyvą bei vektorius (šiuo atveju tai pat masyvas) sudarytą iš jutiklio gautų duomenų. Prieš tai minėtas realiųjų skaičių masyvas yra

užpildomas reikšmėmis pagal vektorių (gautas iš jutiklio). Toliau yra naudojamas *getRotation* metodas, kuris priima realiųjų skaičių masyvą ir 3 realiųjų skaičių dydžio masyvą [4]. Pastarasis yra užpildomas galutinėmis x, y ir z reikšmėmis, kurių prasmingumas yra pavaizduotos 2 iliustracijoje.

2.3. Magnetinio lauko jutiklis ir akselerometras

Magnetinio lauko jutiklis (angl. Magnetic field sensor) yra fizinis jutiklis, kuris matuoja aplinkos magnetinius lauką x, y ir z ašyse. Gautos reikšmės yra nurodytos mikro Tesla (uT) matavimo vienetu. Akselerometras (angl. Accelerometer) tai pat yra fizinis jutiklis, kuris matuoja įrenginiui suteiką akseleraciją. Gauti duomenys yra akseleracija į x, y ir z ašis [3]. Tiek magnetinio lauko jutiklio duomenys, tiek akselerometro duomenys nėra paruošti naudojimui. Visų pirma yra reikalingas nuokrypio išlyginimas. Tam yra naudojama nuokrypio išlyginimo funkcija, aprašyta 2.1 dalyje. Gauti rezultatai yra paduodami *getRotationMatrix* metodui, kuris tai pat yra integruotas į Android operacinę sistemą [4]. Metodas priima 9 arba 16 realiųjų skaičių dydžio masyvą, kuris yra aprašytas 2.2 dalyje. Metodas tai pat priima tokį pat masyvą, kuris yra užpildomas duomenimis, kurie nurodo palenkimą, tačiau šie duomenys nebus reikalingi, todėl jie yra ignoruojami. Toliau yra naudojamas *getRotation* metodas, kuris buvo aprašytas 2.1 skyriuje. Metodas tokiu pat būdu užpildo sukimosi masyvą, kuris turi reikšmes, pavaizduotas 2 iliustracijoje.

2.4. Posūkio nustatymas

Naudojant duomenis, gautus naudojant metodus aprašytus 2.2 ir 2.3 dalyse, galime gauti tikslius laipsnius, nurodančius įrenginio poziciją erdvėje. Android operacinė sistema turi metodą, kuris grąžina vieną iš įrenginio pozicijų: 0°, 90°, 180°, 270° [1]. Šios fiksuotos pozicijos yra naudojamos grafinės vartotojo sąsajos atvaizdavimui. Jeigu reikšmė yra 90° arba 270° ir įrenginyje įjungta vaizdo pavertimo horizontaliai funkcija, grafinė vartotojo sąsaja bus pasukta horizontaliai. Vaizdas grįš į pradinę vertikalią padėtį, kai reikšmė bus 0° arba 180°. Naudojant šią reikšmę galime gauti tikslius laipsnius, kaip įrenginys yra pasuktas. Tokiu būdu yra išgaunami tikslūs kompas laipsniai.

3. Kompiuterio mokymas

Kompiuterio mokymas (angl. Machine learning) yra dirbtinio intelekto sritis, kuri moko kompiuterį nuspėti reikšmes pagal turimus duomenis. Tai yra programų kūrimo būdas, kai sukurta sistema prisitaiko prie duomenų, juos naudojant spėja reikšmes. Kompiuterio mokymas yra stipriai susijęs su statistika, nes kompiuterio mokymas ir statistika nagrinėja duomenų analizę, bet skirtingai nuo statistikos, kompiuterio mokymas yra susijęs su skaičiavimams naudojamų algoritmų sudėtingumais [29]. Dažnu atveju, sistemos mokymasis nesugeba gauti visiškai tikslaus rezultato, todėl toks būdas yra geriausias sprendimams, kuriuose nėra reikalingas idealus tikslumas. Kaip ir visiems gerai žinomoje globalioje padėties nustatymo sistemoje (angl. Global positioning system), taip ir navigacijai patalpose, idealus tikslumas nėra būtinas, tačiau kuo didesnis tikslumas yra didelis privalumas.

3.1. Kompiuterio mokymo modelis

Visų pirma, yra svarbu suprasti kas yra modelis kompiuterio mokymo kontekste. Modeliu (angl. Model) vadinsime tai, ką gauname apmokę algoritmą su tam tikru duomenų rinkiniu. Gautą modelį toliau naudojame prognozuoti atsakymus pagal dar nematytus duomenis. Naudojant tą patį algoritmą su skirtingais duomenimis gausime skirtingus modelius. Naudojant tuos pačius duomenis su skirtingais algoritmais tai pat gausime skirtingus modelius. Modelio unikalumas tiesiogiai priklauso nuo pasirinkto algoritmo ir turimų duomenų.

3.2. Kompiuterio mokymo metodai

Kompiuterio mokymas nėra tikslusis mokslas. Tai apima platų kompiuterio mokymo įrankių, metodų ir idėjų spektrą. Šiuo metu dažniausiai yra sutinkamos trys skirtingos technikos naudojamos kompiuterio mokymui.

3.2.1. Prižiūrimas mokymas

Prižiūrimas mokymas (angl. Supervised learning) yra kompiuterio mokymo užduotis sudaryti modelį, pagal kurį įvestis ir išvestis susiejama remiantis įvesties ir išvesties porų pavyzdžiais. Toks mokymas daro išvadą iš turimų mokymo duomenų, kuri yra sudaryta iš mokymo pavyzdžių [15]. Toks pavyzdys susidaro iš įvesties (dažniausiai duomenų vektorius) ir norimos išvesties (tokia išvestis tai pat yra vadinama priežiūros signalu). Prižiūrimo mokymo algoritmas analizuoja duomenis ir sudaro modelį, kuris gali būti naudojamas naujiems pavyzdžiams susieti. Optimaliu scenarijumi, algoritmas sugebės sudaryti modelį, kuris galės išgauti tinkamą išvestį naudojant dar nematytą įvestį. Tokiam mokymui yra reikalingi duomenys, turintys žymeklius (angl. labels).

3.2.2. Neprižiūrimas mokymas

Neprižiūrimas mokymas (angl. Unsupervised learning) yra mokymosi tipas, kuomet duomenys nėra suklasifikuoti ir neturi žymeklių [28]. Tokio mokymo tikslas yra sumodeliuoti pamatinę duomenų struktūrą arba pasiskirstymą duomenyse. Taip yra sugrupuojami duomenys ir yra išskirti į blokinius, kurie turi panašių savybių. Neprižiūrimame mokyme nėra teisingų atsakymų iš

kurių algoritmas galėtų mokytis, todėl teisingus atsakymus išskiria pats modelis, kurį pagal turimus duomenis sudarė algoritmas. Duomenų vizualizacija tai pat gali būti laikoma neprižiūrimo mokymo tipu.

3.2.3. Skatinamasis mokymas

Skatinamasis mokymas (angl. Reinforcement learning) yra mokymosi tipas, kada yra pasirinkami atitinkamai veiksmai siekiant maksimaliai padidinti atlygį konkrečioje situacijoje [25]. Jį naudoja įvairios programinės įrangos ir mašinų sistemos, norinčios rasti geriausią įmanomą elgesį ar kelią, kurį turėtų pasirinkti konkrečioje situacijoje. Skatinamasis mokymas nuo prižiūrimo mokymo tuo, jog prižiūrimo mokymo metu mokymo duomenys turi įvestį su teisinga išvestimi, dėl to tokio mokymo metu algoritmas žino dalį teisingų atsakymų. Skatinamojo mokymo metu skatinamasis agentas pats nusprendžia teisingą atsakymą ir bando pasirinkti teisingą atsakymo gavimo būdą. Mokymo duomenų nebuvimas priverčia skatinamąjį mokymąsi naudojančią sistemą mokytis iš savo patirties.

3.3. Duomenų dalinimas

Prižiūrimam mokymui visų pirma duomenys turėtų būti padalinami į dvi dalis. Remiantis Pareto principu, mokymo duomenys dažniausiai sudaro 80% visų duomenų, o likę 20% - testavimo duomenys. Duomenys turėtų būti išskirstomi atsitiktine tvarka. Kai kuriais atvejais 80%/20% santykis nėra optimaliausias. Tokiu atveju, duomenys turėtų būti išskirstomi kitaip, tačiau santykis turėtų išlikti tarp 60%/40% ir 90%/10%. Didesnis arba mažesnis santykis neleis algoritmui teisingai sudaryti funkcijos arba funkcija nebus ištestuota teisingai. Neprižiūrimam mokymui bei skatinamajam mokymui duomenų dalinimas dažnu atveju nėra reikalingas.

3.4. Duomenų normalizacija

Duomenų normalizacija (angl. Normalization) yra duomenų apdorojimo metodas, kuomet duomenys yra suvienodinami [19]. Tai yra svarbu todėl, nes daugelis kompiuterio mokymo algoritmų ir metodų skaičiuoja atstumą tarp dviejų taškų naudojant euklidinį atstumą. Jeigu kažkuris duomenų stulpelis turi didelį diapozoną, atstumas, lyginant su kitu duomenų stulpeliu, kuris turi mažesnį diapozoną, bus neproporcingai skirtingas. Dėl šios priežasties, didesnį diapozoną turintis stulpelis turės didesnę įtaką lyginant su mažesnį diapozoną turinčiu stulpeliu. Todėl visų stulpelių diapozonas turėtų būti normalizuotas, kad kiekvieno stulpelio euklidinis atstumas turėtų vienodus didžiausius ir mažiausius atstumus.

Vienas iš dažnai naudojamų būdų yra min-max perskaičiavimas [19]. Tai yra paprasčiausias normalizacijos metodas kuris susideda iš diapozono pakeitimo iš esamo į [0, 1] arba [-1, 1]. Diapozono pasirinkimas stipriai priklauso nuo duomenų pobūdžio. Bendroji formulė, pakeičianti diapozoną į [0, 1]

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (3.1)$$

kur x yra originali reikšmė, o x' normalizuota reikšmė. Pavyzdžiui, tarkime, kad turime mėnesinių algų duomenis, kurių diapozonas yra [400, 2000] eurų. Norint perskaičiuoti šiuos duomenis, iš kiekvienos algos atimame mažiausią turimą algą (400 eurų) ir rezultatą padaliname iš didžiausios ir mažiausios algų skirtumo (1600 eurų). Gautas rezultatas visada bus [0, 1] diapozone. Bendroji

formulę, pakeičianti reikšmes pasirinktame $[a, b]$ diapozone

$$x' = a + \frac{(x - \min(x))(a - b)}{\max(x) - \min(x)} \quad (3.2)$$

kur a ir b ir mažiausia ir didžiausia reikšmės pasirinktame diapozone. Naudojant šią formulę su pasirinktu diapozonu $[-1, 1]$, gauname reikšmes prieš tai minėtame diapozone, kuris yra identiškas pasirinktam.

3.5. Klasifikacija

Klasifikacija (angl. Classification) yra naudojama tada, kai yra reikalingi kategoriniai rezultatai [21]. Klasifikacijos algoritmai pagal duotą įvestį sukuria funkciją (modelį), kuri grąžina kategorinius rezultatus. Pavyzdžiui, visi elektroniniai laišškai gali būti suskirstyti į dvi klases - šlamštą ir ne šlamštą. Klasifikavimo modeliams yra įprasta numatyti testinę rezultato reikšmę ir pagal ją nuspręsti, kuriai klasei priklauso gautas rezultatas. Gauti rezultatai gali būti interpretuojami kaip tikimybė, pagal kurią yra parenkama konkreti klasė. Pavyzdžiui, konkrečiam elektroniniam laiškui gali būti priskirta 10% (0.1) tikimybė, jog tai yra šlamštas ir 90% (0.9) tikimybė, kad tai nėra šlamštas. Šiais tikimybės galima paversti į vieną iš dviejų klasių. Pagal turimas tikimybės, šį elektroninį laišką galima klasifikuoti kaip ne šlamštą. Klasifikavimo modelio tikslumas gali būti skaičiuojamas keliais būdais, tačiau vienas iš dažniausių ir paprasčiausių yra paskaičiuoti modelio teisingai nuspėtų reikšmių procentą [21]. Pavyzdžiui, iš 5 laiškų, kurie visi buvo šlamštas, 3 buvo klasifikuoti kaip šlamštas ir 2 kaip ne šlamštas. Tokiu atveju, modelio tikslumas yra 60% ($\frac{3}{5}$). Bendra formulė

$$x = \frac{a}{a + b} \quad (3.3)$$

kur x yra tikimybė, a yra teisingai nuspėtos klasės ir b - neteisingai nuspėtos klasės

Toliau yra apžvelgiami 2 klasifikacijos metodai, kurie bus naudojami tyrimo metu.

3.5.1. Naiviojo Bajeso klasifikatorius

Naiviojo Bajeso (angl. Naive Bayes) yra klasifikavimo metodas, paremtas Bajeso teorema su prielaida, jog bruožai (duomenys) yra nepriklausomi vienas nuo kito [5]. Pavyzdžiui, vaisus gali būti laikomas obuoliu, jeigu jis yra raudonas, apvalus ir 7 centimetrų skersmens. Net jei šie bruožai tiesiogiai priklauso vienas nuo kito arba nuo kitų savybių buvimo, visi šie bruožai nepriklausomai prisideda prie tikimybės, kad šis vaisius iš tikrųjų yra obuolys. Šis klasifikatorius yra itin populiarus dėl to, kad modelis yra pakankamai paprastai sukonstruojamas ir gali būti naudojamas su nedideliu duomenų kiekiu [5]. Bajeso teorema suteikia galimybę apskaičiuoti A hipotezės galimybę atsirandant įvykiui B.

$$Pr(A|B) = \frac{Pr(B|A)Pr(A)}{Pr(B)} \quad (3.4)$$

kur

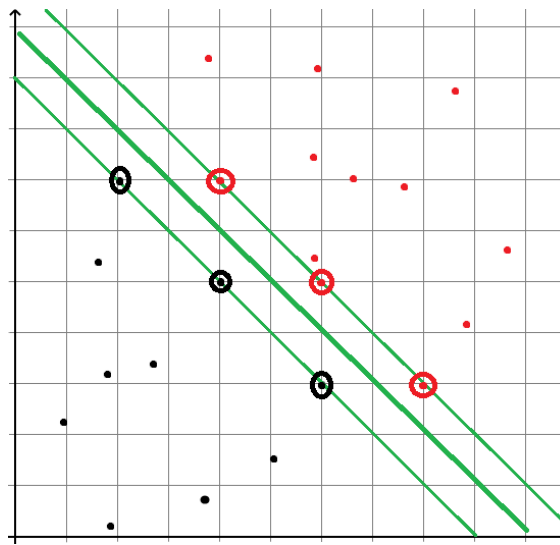
- $Pr(A|B)$ yra A hipotezės tikimybė, atsirandant B įvykiui;
- $Pr(B|A)$ yra įvykio B atsiradimo tikimybė esant teisingai A hipotezei;
- $Pr(A)$ yra apriorinė (nepriklausoma nuo patirties) A hipotezės tikimybė

- $\Pr(B)$ yra įvykio B tikimybė.

Prižiūrimam kompiuterio mokymui yra naudojamas Gauso naiviojo Bajeso (angl. Gaussian Naive Bayes) klasifikacijos algoritmas. Šis algoritmas daro prielaidą, jog duomenys yra pasiskirstę pagal Gauso kreivę. Šis algoritmas yra itin populiarus dėl savo paprastumo ir našumo, lyginant su kitais klasifikavimo algoritmais [27].

3.5.2. Atraminių vektorių klasifikatorius

Atraminių vektorių klasifikatorius (angl. Support vector machine) yra klasifikavimo metodas ir gali būti naudojamas prižiūrimam kompiuterio mokymui [23]. Atraminių vektorių klasifikatoriaus metodas siekia surasti optimaliausią hiperplokštumą (angl. hyperplane) tarp klasių sutelkiant dėmesį į atvejus, kai duomenys ribiniai klasės atvejai. Šie atvejai yra vadinami atraminiais vektoriais. Tai geriau suprasti padės 3 iliustracija. Kiti atvejai, kurie nėra išsidėstę klasės pakraščiuose,



3 pav. Atraminių vektorių radimas. Apskritimais pažymėti taškai yra atraminiai vektoriai. Raudoni ir juodi taškai nurodo dvi skirtingas klases.

yra pašalinami iš mokymo duomenų. Dėl šios priežasties, likę treniravimo duomenys yra gerokai naudingesni lyginant su originaliais treniravimo duomenimis. Likę duomenys nurodo tik tikslas klasifikavimo ribas, jų yra gerokai mažiau. Metodas iš turimų duomenų sugeba sukurti klasifikavimo modelį. Dėl mažo duomenų kiekio ir metodo pobūdžio, modelis veikia našiai [23].

3.6. Regresija

Regresija (angl. Regression) yra naudojama tada, kai yra reikalingas tęstinis rezultatas. Regresijos algoritmai pagal duotą įvestį sukuria funkciją (modelį), kuri gražina faktines reikšmes. Tokios reikšmės gali būti sveikieji, realieji skaičiai. Pavyzdžiui, žmonių kiekis restorane gali svyruoti nuo 0 iki 100 vakaro metu. Taigi, regresijos algoritmai turėtų būti naudojami tada, kai yra reikalingas kiekis.

Kadangi reikšmės yra kiekinės, modelio tikslumas turėtų būti skaičiuojamas kaip klaidos dydis tarp gauto ir tikėtino rezultatų. Yra keletas būdų įvertinti regresinio modelio įgūdžius, vienas iš jų yra vidutinio kvadratinio nuokrypio skaičiavimas [17].

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (Xm_i - Xk_i)^2}{n}} \quad (3.5)$$

kur Xm_i yra teisinga reikšmė, Xk_i yra nuspėta reikšmė, n yra matavimų skaičius.

Pavyzdžiui, modelis atliko 2 spėjimus. Vieno spėjimo rezultatas yra 2,5, teisinga reikšmė - 2, kito spėjimo rezultatas - 4, teisinga reikšmė - 3,3. Norint paskaičiuoti modelio įgūdžius, reikia panaudoti 3.5 formulę.

$$\sigma = \sqrt{\frac{(2 - 2,5)^2 + (3,3 - 4)^2}{2}} = 0.37 \quad (3.6)$$

Mažesnis nuokrypis reiškia geresnius modelio įgūdžius nuspėti rezultatui. Dėl skaičiavimo pobūdžio, didesnis netikslumas reikšmingai pakeičia gautą rezultatą.

Kitas būdas modelio efektyvumo įvertinimui yra determinacijos koeficiento (toliau R^2) skaičiavimas [18]. R^2 yra priklausomo kintamojo dispersijos dalis, kurią galima nuspėti iš nepriklausomų kintamųjų. Bendroji formulė

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad (3.7)$$

kur SS_{res} yra likutinė kvadratų suma, SS_{tot} bendra kvadratų suma.

3.6.1. K-artimiausio kaimyno regresija

K-artimiausio kaimyno (angl. K-nearest neighbour) algoritmas yra neparametrinis metodas naudojamas tiek regresijai, tiek klasifikacijai. Metodas pirmą kartą buvo pristatytas [13] leidinyje ir galutinai įformintas klasifikacijos užduotims vykdyti [9]. Anot Demand Fahad H. Al-Qahtani ir Sven F. Crone, k-artimiausio kaimyno algoritmas gali būti naudojamas ir regresijai [11]. Abiem atvejais, įvestį sudaro k artimiausių mokymosi pavyzdžiai bruožų erdvėje. Naudojant šį algoritmą regresijai, išvestis yra k artimiausių kaimynų sumos vidurkis. Tai pat, artimiausi kaimynai gali turėti "svorį" pagal tai, koku euklidiniu atstumu yra nuo tiriamojo taško [11]. Dažniausiai svoris yra skaičiuojamas $\frac{1}{d}$, kur d yra euklidinis atstumas iki kaimyno.

3.6.2. Ridge regresija

Ridge regresija (angl. Ridge regression), dar kitaip žinoma kaip Tichonovo reguliarizacija (angl. Tikhonov regularization), yra regresijos metodas, kuris yra skirtas tirti duomenims, kurie kenčia nuo daugialypiškumo (angl. multicollinearity) [16]. Šis metodas gali būti naudojamas ir kompiuterio mokymo regresijos problemoms spręsti [16][2]. Esant daugialypiams duomenis, lyginant su mažiausių kvadratų (angl. Least squares) regresijos metodu, Ridge regresija yra pranašesnė. Tai pasireiškia koeficientų keitimu priklausomai nuo jų dydžio. Taip pakeisti koeficientai yra atsparesni daugialypiškumui [2]. Ši savybė gali būti naudinga tiriant RSSI indikatorius reikšmes. Du arba daugiau BLE švyturėlio kanalai gali siųsti signalą panašiu stiprumu, dėl to RSSI indikatorius reikšmės, gautos iš minėtų kanalų, gali tapti daugialypėmis. Šis metodas yra naudojamas spręsti panašią lokalizacijos problemą, kuri tai pat naudoja RSSI indikatorius reikšmes[20].

4. Tyrimas

Tyrimo tikslas yra sužinoti, ar navigacijai patalpose, kuri naudoja iš BLE stotelių užfiksuotus RSSI indikatoriaus duomenis, kompasu duomenys bei duomenų normalizavimas daro teigiamą įtaką naudojant kompiuterio mokymo klasifikacijos ir regresijos metodus. Yra tikimasi, jog klasifikacija bus tikslesnė, o regresija turės geresnes R^2 ir RMSE reikšmes.

4.1. Duomenys ir jų rinkimas

Šiam tyrimui buvo pasirinktas Didlaukio MIF patalpų trečias aukštas. Tyrimo metu buvo naudojami jau surinkti ir apdoroti BLE stotelių duomenys atitinkamuose koordinatinių taškuose. Šie duomenys yra viešai prieinami Doc., Dr. Valdo Rapševičiaus Git repozitorijoje [12], esančioje "Github" versijų sistemoje. Be duomenų, repozitorijoje yra programėlės, skirtos Android operacinei sistemai, išeities kodas bei skriptai, skirti duomenų apdorojimui. Aplikacija yra naudojama duomenų rinkimui, kurie yra reikalingi piršto antspaudo metodui įgyvendinti. Metodas yra aprašytas 1.4.2 skyriuje.

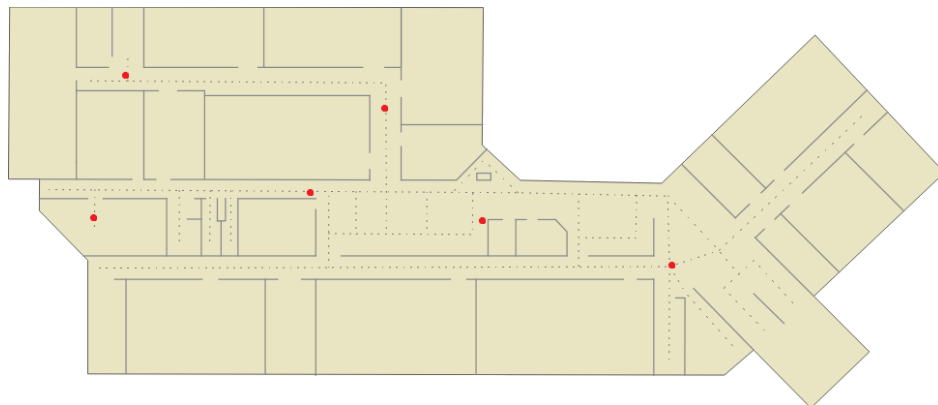
4.1.1. BLE stotelių duomenys

Programėlė renka dviejų tipų duomenis. Vieno tipo duomenys yra susiję su koordinatinių taškais. Juos sudaro koordinatinių taško identifikatorius, laiko žyma, geografinė platuma bei geografinė ilguma. Duomenų pavyzdys yra matomas 1 lentelėje.

Identifikatorius	Laiko žyma	Geografinė platuma	Geografinė ilguma
165	1561025163188	54.72986343	25.26270544

1 lentelė. Android programėlės surinktų duomenų koordinatinių taškui pavyzdys

Kito tipo duomenis sudaro laiko žyma, BLE stotelės tinklo plokštės adresas, kanalo identifikatorius ir RSSI reikšmė. RSSI indikatoriaus reikšmės yra gaunamos iš žemėlapyje, pavaizduoto 4 iliustracijoje, pažymėtų 6 BLE stotelių 36 skirtingų kanalų (kiekviena stotelė turi 6 kanalus). Šio



4 pav. Raudonais taškais pažymėtos BLE stotelių pozicijos.

tipo duomenų pavyzdys yra matomas 2 lentelėje.

Laiko žyma	BLE stotelės tinklo plokštės adresas	Kanalo identifikatorius	RSSI
1561028666797	E5:53:1E:24:96:2F	3af9ac1c-c3a5-36f7-a737-4868789e2832	-92

2 lentelė. Android programėlės surinktų duomenų RSSI reikšmių pavyzdys

Koordinatinių taškų žemėlapis yra matomas 5 iliustracijoje. Šis žemėlapis yra paimtas tiesiai iš programėlės. Paspaudus ant bet kurio iš 165 taškų žemėlapyje, tuo metu yra nuskaitomos taško geografinės koordinatės, taško identifikatorius ir tiksli laiko žyma, kada buvo žymeklis buvo paspaustas.



5 pav. 3 aukšto planas su koordinatinių taškais.

RSSI indikatoriaus reikšmės yra renkamos įjungus skenavimo režimą. Skenavimo metu yra saugomos visos išmanųjų Android įrenginių pasiekę RSSI indikatoriaus reikšmės. Kiekvienai reikšmei tai pat yra priskiriama laiko žyma, BLE stotelės tinklo plokštės adresas ir kanalo identifikatorius.

Šie duomenys nėra paruošti piršto antspaudų žemėlapiui sudaryti. Jie yra surišami naudojant laiko žymas. Repozitorijoje [12] yra randamas skriptas, kuris apjungia šiuos duomenis. Prie koordinatinių taškų duomenų yra prijungiamos visos RSSI indikatoriaus reikšmės, kurios patenka į aprašytą laiko intervalą. Rezultato formatas yra papildyta 1 lentelė. Prie lentelės yra pridedama 36 papildomi stulpeliai su RSSI indikatoriaus reikšmėmis. Papildomų stulpelių skaičių sudaro 6 BLE stotelių 36 skirtingų kanalų RSSI indikatoriaus reikšmės. Jeigu duomenų iš konkretaus kanalo nėra, reikšmė yra užpildoma 0. Stulpelių pavadinimus sudaro unikalios BLE stotelių tinklo plokštės ir kanalo identifikatoriaus kombinacijos. Duomenų rinkinį sudaro 1426 duomenų eilutės. Tokio duomenų skaičiaus turėtų užtekti efektyviam kompiuterio mokymui.

4.1.2. Kompasso duomenys

Sekantis žingsnis yra kompasso duomenų gavimas ir pridėjimas prie esančių duomenų. Kompasso duomenų gavimui yra reikalingi programėlės pakeitimai. Naudojantis informacija, pateikta 2 skyriuje, buvo įgyvendinti programėlės pakeitimai. Dabar, prie koordinatinių taškų lentelės yra pridedamas kompasso stulpelis. Reikšmė yra taško paspaudimo metu vėliausiai išsaugoti kompasso

laipsniai. Kompasso reikšmė yra skaičiuojama daug kartų per sekundę ir kiekvieno paskaičiavimo rezultatas yra įrašomas kaip vėliausiai išsaugota reikšmė.

Reikalingų jutiklių duomenys buvo renkami naudojant "POCOPHONE F1" išmanųjį telefoną, kuris naudoja Android 9 operacinę sistemą. Dėl duomenų vientisumo, duomenų rinkimo metu buvo stengiamasi laikyti išmanų telefoną taip, jog jo nuolydis ir pasisukimas nekistų. Tai buvo įgyvendinta telefoną nešant atvirame delne, stengiantis išlaikyti tokią pat delno poziciją. Pasisukus visam kūnui pasikeičia ir telefono posūkio parametras. Išlaikant vienodą nuolydį ir pasisukimą, posūkio parametras keisis pagal pasisukimo laipsnį. Duomenų rinkėjui pasisukus 90° kampu, telefono posūkio parametras tai pat nukryps maždaug per 90° į kažkurią pusę. Kompasso reikšmė nurodo pasisukimo kampą, todėl visada išliks tarp 0° ir 360° .

Surinkti kompasso duomenys yra prijungiami prie turimų duomenų pagal koordinačių taško identifikatorių. Turimų duomenų aibė yra didesnė lyginant su kompasso duomenimis. Dėl šios priežasties, kiekvienai turimai duomenų eilutei, turinčiai vienodą identifikatorių, parinksime vieną iš tame taške surinktų kompasso duomenų reikšmę. Toliau darysime prielaidą, jog pro koordinačių tašką yra einama arba pirmyn, arba atgal. Dėl to, kas antram taškui, turinčiam identišką identifikatorių, originalią reikšmę pasuksime 180° kampu. Tokiu būdu, maždaug pusė eilučių, turinčių vienodą identifikatorių, turės kompasso laipsnius, rodančius priešingą pusę lyginant su originalia reikšme. Jog duomenys labiau atitiktų realybę, kiekvieną originalią ar 180° pasuktą kompasso reikšmę pasuksime laipsniu iš aibės $[-10^\circ, 10^\circ]$. Tokiu būdu gausime nuokrypį, kuris atspindės faktą, jog kiekvieną kartą telefono posūkis nebus vienodas, nors ir telefonas bus nukreiptas į tą pačią pusę.

4.2. Kompiuterio mokymo eiga ir rezultatai

Tyrimo metu buvo tiriami tiek klasifikacijos, tiek regresijos metodai. Klasifikacijai buvo naudojami Gauso naiviojo Bajeso (toliau GNB), Atraminų vektorių (toliau SVM) ir K-artimiausio kaimyno (toliau KNN) klasifikatoriai. Regresijos problemai spręsti buvo naudojami KNN ir Ridge regresijos metodai. Visais atvejais duomenys buvo dalinami į dvi dalis: mokymo - 80% ir testavimo - 20%. Visi naudojami algoritmai buvo naudojami su numatytais parametrais. Klasifikacijos tikslumas buvo skaičiuojamas 3.5 skyriuje aprašytu metodu. Regresijos tikslumui matuoti buvo naudojamas vidutinis kvadratinis nuokrypis (toliau. RMSE) ir R^2 . Klasifikacijos ir regresijos modelių sudarymui ir tikslumo matavimui buvo naudojama Python programavimo kalbai pritaikyta kompiuterio mokymui skirta "scikit-learn" biblioteka. Klasifikacijai buvo parinktos 165 skirtingos klasės, kurios yra koordinačių taškų identifikatoriai. Tai reiškia, jog klasifikacijos modelis turės nuspėti, kuriam taškui priklauso tiriami duomenys. Regresijos metu buvo nuspėjamos tikslios platumos ir ilgumos reikšmės.

Toliau bus atliekami keturi tyrimai su skirtingais ir skirtingai apdorotais duomenimis.

- Pirmas tyrimas: naudojamos RSSI indikatoriaus reikšmės;
- Antras tyrimas: naudojamos RSSI indikatoriaus ir kompasso reikšmės;
- Trečias tyrimas: naudojamos normalizuotos RSSI indikatoriaus reikšmės;
- Ketvirtas tyrimas: naudojamos normalizuotos RSSI indikatoriaus ir kompasso reikšmės.

Iš tyrimų rezultatų yra tikimasi dviejų dalykų:

- Kompasso reikšmės pagerins tiek klasifikacijos, tiek regresijos modelių spėjimų tikslumą;

- Duomenų normalizavimas pagerins tiek klasifikacijos, tiek regresijos modelių spėjimų tikslumą.

Metodas	Tikslumas
GNB klasifikacija	26.92%
SVM klasifikacija	25.17%
KNN klasifikacija	25.52%

3 lentelė. Kompiuterio mokymo klasifikacijos metodų, naudojančių BLE stotelių RSSI indikatoriaus reikšmes, prognozių tikslumas.

Metodas	R^2 (ilg.)	RMSE (ilg.)	R^2 (plat.)	RMSE (plat.)
KNN regresija	0.968	4.832^{-5}	0.728	2.16^{-5}
Ridge regresija	0.875	9.6^{-5}	0.556	2.758^{-5}

4 lentelė. Kompiuterio mokymo regresijos metodų, naudojančių BLE stotelių RSSI indikatoriaus reikšmes, R^2 ir RMSE reikšmės.

3 ir 4 lentelėse yra pavaizduoti pirmojo tyrimo rezultatai. Tyrimo metu, modelių apmokymui buvo naudojamos tik RSSI indikatorių reikšmės.

Iš klasifikacijos modelių geriausią rezultatą sugebėjo gauti GNB klasifikacijos modelis. Nedideliu skirtumu atsiliko SVM ir KNN klasifikacijos modeliai.

Iš regresijos modelių tiek ilgumą, tiek platumą tiksliausiai spėjo KNN regresijos modelis. Lyginant su Ridge regresijos modeliu, ilgumos R^2 skirtumas nėra didelis. Su platuma abiem regresijos modeliams susitvarkyti pavyko blogiau. R^2 reikšmių skirtumas yra nemažas, Ridge regresijos modeliui platumą sekėsi gerokai blogiau.

4.2.1. Kompasų duomenų įtaka

5 ir 6 lentelėse yra pavaizduoti antrojo tyrimo rezultatai. Šio tyrimo metu, prie modelių apmokymo duomenų prisidėjo kompasų reikšmės.

Metodas	Tikslumas
GNB klasifikacija	26.92%
SVM klasifikacija	23.08%
KNN klasifikacija	21.33%

5 lentelė. Kompiuterio mokymo klasifikacijos metodų, naudojančių BLE stotelių RSSI indikatoriaus ir kompasų reikšmes, prognozių tikslumas.

Iš klasifikacijos modelių geriausią rezultatą sugebėjo gauti GNB klasifikacijos modelis. Jau didesniu skirtumu atsiliko SVM ir KNN klasifikacijos modeliai. Lyginant su pirmu bandymu, SVM ir KNN klasifikacijos modelių rezultatai suprastėjo.

Tarp regresijos modelių vėl geresnius rezultatus gavo KNN regresijos modelis. Lyginant su pirmojo tyrimo rezultatais, KNN regresijos modeliui ilgumą nuspėti sekėsi šiek tiek prasčiau, tačiau platumos spėjimas buvo žymiai geresnis. Ridge regresijos modelio rezultatai pakito neženkliai į teigiamą pusę, prognozuojant tiek ilgumą, tiek platumą, rezultatai pagerėjo.

Metodas	R^2 (ilg.)	RMSE (ilg.)	R^2 (plat.)	RMSE (plat.)
KNN regresija	0.965	5.096^{-5}	0.807	1.817^{-5}
Ridge regresija	0.875	9.596^{-5}	0.562	2.739^{-5}

6 lentelė. Kompiuterio mokymo regresijos metodų, naudojančių BLE stotelių RSSI indikatorius ir kompasų reikšmes, R^2 ir RMSE reikšmės.

4.2.2. Duomenų normalizavimo įtaka

Duomenų normalizavimui buvo taikomas min-max perskaičiavimas. Kaip ir buvo minėta 3.4 skyriuje, min-max metodas perskaičiuoja duomenis taip, kad jų diapozonas pasikeistų į [0, 1] arba į [-1, 1]. Šiuo atveju buvo pasirinktas [0, 1] diapozonas, buvo perskaičiuotos tiek RSSI indikatorius, tiek kompasų reikšmės.

7 ir 8 lentelėse yra pavaizduoti trečiojo tyrimo rezultatai. Modelių apmokymui buvo naudojamos normalizuotos RSSI indikatorius reikšmės.

Metodas	Tikslumas
GNB klasifikacija	26.92%
SVM klasifikacija	25.87%
KNN klasifikacija	26.57%

7 lentelė. Kompiuterio mokymo klasifikacijos metodų, naudojančių BLE stotelių RSSI indikatorius normalizuotas reikšmes, prognozių tikslumas.

Metodas	R^2 (ilg.)	RMSE (ilg.)	R^2 (plat.)	RMSE (plat.)
KNN regresija	0.968	4.861^{-5}	0.731	2.146^{-5}
Ridge regresija	0.875	9.605^{-5}	0.557	2.755^{-5}

8 lentelė. Kompiuterio mokymo regresijos metodų, naudojančių BLE stotelių RSSI indikatorius normalizuotas reikšmes, R^2 ir RMSE reikšmės.

Tarp klasifikacijos modelių, kaip ir praeituose tyrimuose, geriausią rezultatą rodo GNB klasifikacijos modelis. KNN klasifikacijos ir SVM klasifikacijos modeliai gavo geresnius rezultatus lyginant su pirmu tyrimu.

Lyginant su pirmu tyrimu, KNN regresijos modelio ilgumos spėjimas pamažėjo, tačiau nereikšmingai, o ilgumos - neženkliai padidėjo. Ridge regresijos modelio ilgumos spėjimai sumažėjo nereikšmingai, o ilgumos nereikšmingai padidėjo.

9 ir 10 lentelėse yra pavaizduoti ketvirtojo tyrimo rezultatai. Modelių apmokymui buvo naudojamos normalizuotos RSSI indikatorius ir kompasų reikšmės.

Tarp klasifikacijos modelių šį kartą geriausiai duomenis spėjo KNN klasifikacijos modelis. Visus praeitus tyrimus pirmavęs ir tokį pat spėjimo procentą išlaikęs GNB klasifikacijos modelis šio tyrimo metu spėjo šiek tiek prasčiau. SVM klasifikatoriaus modelis nedaug atsiliko nuo kitų klasifikacijos modelių, tačiau pagal spėjimų taiklumą liko paskutinis. Įdomu tai, kad stipriai padidėjo KNN klasifikacijos ir SVM klasifikacijos modelių teisingų spėjimų procentas lyginant su antru tyrimu. Šio tyrimo metu, visi klasifikacijos modeliai pasiekė labai panašų teisingų spėjimų procentą.

Metodas	Tikslumas
GNB klasifikacija	26.57%
SVM klasifikacija	26.22%
KNN klasifikacija	26.92%

9 lentelė. Kompiuterio mokymo klasifikacijos metodų, naudojančių BLE stotelių RSSI indikatorius ir kompasu normalizuotas reikšmes, prognozių tikslumas.

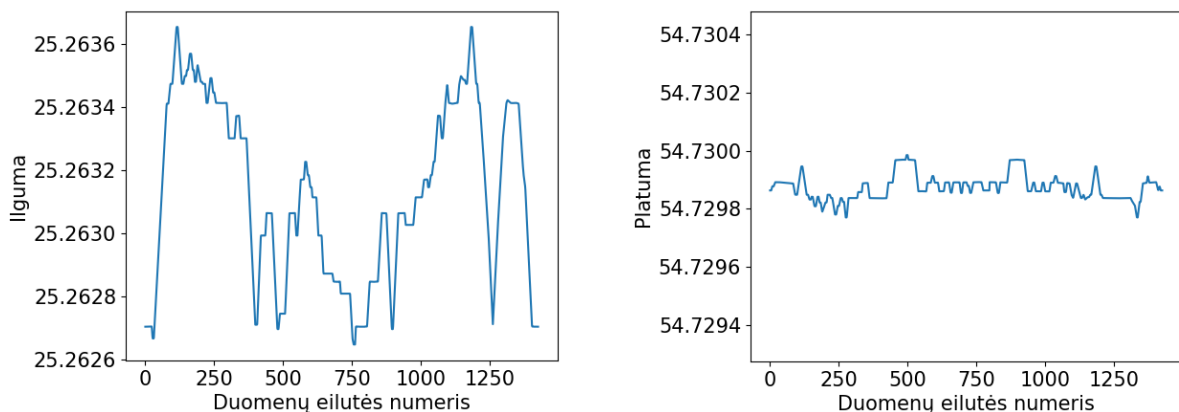
Metodas	R^2 (ilg.)	RMSE (ilg.)	R^2 (plat.)	RMSE (plat.)
KNN regresija	0.965	4.858^{-5}	0.799	1.854^{-5}
Ridge regresija	0.875	9.601^{-5}	0.563	2.736^{-5}

10 lentelė. Kompiuterio mokymo regresijos metodų, naudojančių BLE stotelių RSSI indikatorius ir kompasu normalizuotas reikšmes, R^2 ir RMSE reikšmės.

Lyginant su antrojo tyrimo rezultatais, KNN regresijos modelis platumą ir ilgumą spėjo prasčiau, o Ridge regresijos modelis praktiškai nepakito. Tiek ilgumą, tiek platumą, kaip ir visuose tyrimuose, spėja KNN regresijos modelis.

4.2.3. Regresijos tikslumo rodiklių analizė

Visuose keturiuose tyrimuose galima išvelgti vieną tendenciją, susijusią su regresijos R^2 ir RMSE reikšmėmis. R^2 reikšmės geriausias rezultatas yra 1. Geriausias RMSE reikšmės rezultatas yra 0. Pavyzdžiui, pirmame tyrime, nuspėjant ilgumą, KNN regresijos modelio taiklumas buvo įvertintas $0.968 R^2$ ir 4.832^{-5} RMSE reikšmėmis. Toliau, nuspėjant ilgumą, modelio taiklumas buvo įvertintas $0.728 R^2$ reikšme. Matant ilgumos R^2 ir RMSE bei platumos R^2 reikšmes galime daryti prielaidą, jog, tiriant platumą, RMSE reikšmės įvertinimas bus didesnis negu buvo tiriant ilgumą, tačiau tokia prielaida nėra teisinga, nes RMSE reikšmė yra daugiau nei dvigubai mažesnė. Taip nutinka dėl ilgumos bei platumos duomenų pasiskirstymo. Ilgumos ir platumos didžiausių ir mažiausių reikšmių skirtumas skiriasi daugiau nei 4 kartus. Duomenų pasiskirstymą pagal ilgumos pasiskirstymo diapozoną galima pamatyti 6 iliustracijoje. Kaip yra išsiaiškinta 3.6 skyriuje, RMSE



6 pav. Iliustracija kairėje: ilgumos pasiskirstymas. Iliustracija dešinėje: platumos pasiskirstymas ilgumos pasiskirstymo diapozone.

skaičius priklauso nuo tikrų ir spėjamų reikšmių skirtumų. Pažiūrėjus į 5 žemėlapi matome, jog

ilgumos ir platumos duomenų pasiskirstymas turėtų būti priešingas, kadangi koordinatų taškai yra labiau išdėstyti pagal platumą, o ne ilgumą. Ilgumos ir platumos vienoda skaitinė paklaida neatspindės vienodos paklaidos žemėlapyje. Ilguma visada nukryps daugiau. Dėl šios priežasties, ilgumos ir platumos spėjimų tikslumo lyginimui negalima naudoti RMSE reikšmės, tačiau ją galima naudoti lyginant skirtingų modelių spėjimo tikslumą.

4.3. Tyrimo rezultatai

Tyrimo rezultatuose matome, jog koordinates spėjant klasifikacijos metodais, kompasu duomenys duoda teigiamos įtakos, tačiau tik tuo atveju, kai tiek kompasu, tiek RSSI indikatorių reikšmės yra normalizuojamos. Įdomu tai, jog nenormalizavus duomenų, kompasu duomenys duoda daug neigiamos įtakos naudojant SVM ir KNN klasifikacijos modelius, o GNB klasifikacijos modelio rezultatai išvis nepakinta. Lyginant trečią ir ketvirtą tyrimus matome panašius tikslumo rezultatus, SVM ir KNN klasifikacijos modelių rezultatai pagerėjo 0.35%, tačiau GNB klasifikacijos modeliui rezultatas suprastėjo tuo pačiu 0.35%. Duomenų normalizavimas visais atvejais pagerina spėjimo tikslumą arba jo išvis nepakeičia, todėl duomenys visada turėtų būti normalizuojami. Turint daugiau duomenų reiktų lyginti GNB ir KNN klasifikacijos modelių rezultatus ir analizuoti, ar kompasu duomenys iš tiesų duoda naudos, kadangi dabar tiek su kompasu duomenimis, tiek ir be jų, naudojant skirtingus modelius, gauname tą patį geriausią rezultatą - 26.92% teisingai nuspėtų taškų.

Naudojant regresijos modelius kartu su kompasu duomenis rezultatai nebuvo visiškai tokie, kokių tikėtasi. Nustatant ilgumą, kompasu duomenys sumažino tiek KNN, tiek Ridge regresijos modelių tikslumą, o nustatant platumą, tie patys duomenys pagerino modelių tikslumą. Lyginant abiejų modelių ilgumos ir platumos nustatymo tikslumą matome, jog tiek su kompasu duomenimis, tiek be jų, ilgumos nustatymo tikslumas yra gerokai didesnis. Tokie rezultatai yra dėl to, jog kai keičiasi ilguma, aplinkui yra daugiau švyturėlių. Tai galima išvelgti žiūrint į 4 ir 5 iliustracijas. Švyturėliai yra geriau išdėstyti pagal ilgumą, dėl to ilgumos spėjimas yra tikslesnis. Tai reiškia, jog yra užfiksuojama daugiau RSSI indikatoriaus reikšmių kai keičiasi ilguma. Duomenų normalizavimas teigiamos įtakos nedaro tiek su kompasu duomenimis, tiek be jų. Lyginant KNN ir Ridge regresijos modelius, KNN modelis gerokai efektyviau panaudojo kompasu duomenis nustatant platumą, rezultatai žymiai pagerėjo, Ridge regresijos modelis elgėsi panašiai, tačiau žymaus rezultato padidėjimo nebuvo.

Išvados ir rekomendacijos

Galima daryti išvadą, kad kompasu duomenys SVM ir KNN klasifikacijos modeliuose duoda teigiamos įtakos tik tada, kai visi duomenys yra normalizuojami, tačiau nauda yra labai nedidelė. GNB klasifikacijos modeliui kompasu duomenys naudos visiškai nesuteikia. Geriausias gautas rezultatas yra vienodas tiek su kompasu duomenimis, tiek be jų. Su kompasu duomenimis geriausią rezultatą pasiekia KNN klasifikacijos modelis, o be kompasu duomenų - GNB klasifikacijos modelis. Galutinis geriausias rezultatas yra 26.92% teisingai nuspėtų reikšmių. Naudojant klasifikacijos modelius kartu su normalizuotomis kompasu reikšmėmis, reiktų rinktis KNN klasifikacijos modelį.

Regresijos modeliai iš kompasu duomenų gauna teigiamos įtakos tik tada, kai BLE švyturėliai yra išdėstyti neteisingai arba jų yra per mažai. Kuomet švyturėliai yra gerai išdėstyti, šį kartą ilgumos atžvilgiu, kompasu duomenys naudos neduoda. Bet kuriuo atveju, regresijos modeliams duomenų normalizavimas naudos nesuteikia. Naudojant regresijos modelius, visada reiktų rinktis KNN regresijos modelį.

Buvo daryta prielaida, jog per tašką einama arba pirmyn, arba atgal. Dėl šios priežasties, toks sistemos modelis būtų labiau tinkamas daug ilgų ir siaurų praėjimų turinčioms patalpoms, kadangi tokiu atveju realios kompasu reikšmės atspindės darytą prielaidą.

Norint gauti geresnius rezultatus, yra būtina išdėlioti daugiau švyturėlių ir surinkti daugiau duomenų, kadangi kompiuterio mokymo algoritmams yra reikalingas kuo didesnis duomenų kiekis. Lyginant klasifikavimo ir regresijos modelius, reiktų rinktis regresijos modelius, kadangi nedidelė paklaida ilgumoje ar platumoje nesukels problemų nustatant koordinates, tačiau netikslus taško parinkimas gali sukelti daugiau problemų, kadangi nežinome, ar netiksliai nustatytas taškas nėra kitoje aukšto pusėje.

Ateities tyrimų planas

Kadangi išmanus įrenginys turi daugiau jutiklių, negu buvo aprašyta, jų panaudojimas turėtų didelę įtaką navigacijos tikslumui. Vienas iš būdų galėtų būti Pedestrian Dead Reckoning (toliau PDR) algoritmo įgyvendinimas. PDR algoritmas apskaičiuoja įrenginio poziciją naudojant prieš tai buvusias įrenginio pozicijas. Įtraukus iš šio algoritmo gautus duomenis į šią sistemą yra tikimasi gauti geresnius rezultatus.

Literatūros šaltiniai

- [1] *Display*. <https://developer.android.com/reference/android/view/Display.html>, Žiūrėta 2020-01-02.
- [2] *Ridge regression and classification*. https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression-and-classification/, Žiūrėta: 2020-01-07.
- [3] *Sensor Event*. <https://developer.android.com/reference/android/hardware/SensorEvent.html>, Žiūrėta 2020-01-02.
- [4] *Sensor Manager*. <https://developer.android.com/reference/android/hardware/SensorManager>, Žiūrėta 2020-01-02.
- [5] C. P. Sumathi B. M. Gayathri. An Automated Technique using Gaussian Naïve Bayes Classifier to Classify Breast Cancer. page 2, 2016.
- [6] Ram Dantu Brandon Gozick, Kalyan Pathapati Subbu and Tomyo Maeshiro. Magnetic maps for indoor navigation. 2011.
- [7] Vedat Coskun Busra Ozdenizci and Kerem Ok. Nfc internal: An indoor navigation system. 2015.
- [8] Rama Krishna C. Chhabra R., Verma S. Detecting aggressive driving behavior using mobile smartphone. In *Proceedings of 2nd International Conference on Communication, Computing and Networking*, pages 515–516, 2019.
- [9] T. M. Cover and P. E. Hart. Nearest Neighbor Pattern Classification. 2018.
- [10] Akiba Cufi C. Townsend K. Davidson, R. *Getting Started with Bluetooth Low Energy*. O'Reilly Media, 2014. Chapter 1.
- [11] Sven F. Crone Demand Fahad H. Al-Qahtani. Multivariate k-Nearest Neighbour Regression for Time Series data - a novel Algorithm for Forecasting UK Electricity. pages 2–3.
- [12] Dr. Valdas Rapševičius Doc. Android application for ble navigation in mif baltupiai. <https://github.com/valdasraps/BleNav-MIFBaltupiai/> Žiūrėta: 2020-01-06.
- [13] J. Hodges Jr. E. Fix. Discriminatory analysis, nonparametric discrimination. 1951.
- [14] Ramsey Faragher and Robert Harle. Location Fingerprinting With Bluetooth Low Energy Beacons. page 2, 2015.
- [15] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. page 2, 2007.
- [16] Susan F. Marseken Lambert M. Surhone, Miriam T. Timplendon. *Data Mining: Practical Machine Learning Tools and Techniques*. 2010. Chapter 335.
- [17] A. Goktas O. Yeniay. An empirical comparison of pattern recognition, neural nets, and machine learning classification methods. In *Hacettepe Journal of Mathematics and Statistics Volume 31*, page 104, 2002.

- [18] Olivier Renaud and Maria-Pia Victoria-Feser. An empirical comparison of pattern recognition, neural nets, and machine learning classification methods. volume 140, pages 817–841, 2014.
- [19] B. B. Misra S. C. Nayak and H. S. Behera. Impact of data normalization on stock index forecasting. page 5, 2014.
- [20] Paul Honeine Joumana Farah Hichem Snoussi Sandy Mahfouz, Farah Mourad-Chehade. Ridge regression and kalman filtering for target tracking in wireless sensor networks. 2018.
- [21] WeissIoannis Kapouleas Sholom M. An empirical comparison of pattern recognition, neural nets, and machine learning classification methods. In *Proceedings of the 11th international joint conference on Artificial intelligence - Volume 1*, pages 781 –782, 1989.
- [22] Gunho Sohn Solomon Chan. Indoor localization using wi-fi based fingerprinting and trilateration techniques for lbs applications. 2012.
- [23] Angelos Tzotsos. Support Vector Machine Approach For Object Based Image Analysis. pages 1–2, 2008.
- [24] Angel Rodriguez Ubejd Shala. Indoor Positioning using Sensor-fusion in Android Devices. page 10, 2011.
- [25] Mehdi Mirza Alex Graves Timothy P. Lillicrap Tim Harley David Silver Koray Kavukcuoglu Volodymyr Mnih, Adrià Puigdomènech Badia. Asynchronous methods for deep reinforcement learning. page 2, 2016.
- [26] Yang X. Zhao Y. Liu Y. Cuthbert L. Wang, Y. Bluetooth positioning using RSSI and triangulation methods. pages 4–5, 2013.
- [27] Fan Chen Yixiao Chen Bo Jiang Hua Zhang Wangchao Lou, Xiaoqing Wang. Sequence Based Prediction of DNA-Binding Proteins Based on Hybrid Feature Selection Using Random Forest and Gaussian Naive Bayes. page 9, 2014.
- [28] P. Singh Wenjian Hu, Rajiv R. and Richard T. Scalettar. Discovering phases, phase transitions, and crossovers through unsupervised machine learning: A critical examination. page 2, 2017.
- [29] E. Frank Witten, I.H. *Data Mining: Practical Machine Learning Tools and Techniques*. 2nd edition, 2005.
- [30] Lei Zhang Zheng Zuo, Liang Liu and Yong Fang. Indoor Positioning Based on Bluetooth Low-Energy Beacons Adopting Graph Optimization. pages 3–4, 2018.
- [31] Yang J. Li Y. Qi L. El-Sheimy Zhuang, Y. Smartphone-Based Indoor Localization with Bluetooth Low Energy Beacons. pages 6–10, 2016.